



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/716,916	11/18/2003	Greg Christopher JR.	07844-602001-P555	6410
21876	7590	06/26/2006	EXAMINER	
FISH & RICHARDSON P.C. P.O. Box 1022 MINNEAPOLIS, MN 55440-1022			CHEN, QING	
		ART UNIT	PAPER NUMBER	
		2191		

DATE MAILED: 06/26/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/716,916	CHRISTOPHER, GREG
	Examiner	Art Unit
	Qing Chen	2191

— The MAILING DATE of this communication appears on the cover sheet with the correspondence address —
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 18 November 2003.
 2a) This action is FINAL. 2b) This action is non-final.
 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-24 is/are pending in the application.
 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
 5) Claim(s) _____ is/are allowed.
 6) Claim(s) 1-24 is/are rejected.
 7) Claim(s) _____ is/are objected to.
 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.
 10) The drawing(s) filed on 18 October 2004 is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)	4) <input type="checkbox"/> Interview Summary (PTO-413)
2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)	Paper No(s)/Mail Date. _____ .
3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____ .	5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)
	6) <input type="checkbox"/> Other: _____ .

DETAILED ACTION

1. This is the initial Office action based on the application filed on November 18, 2003.

Claims 1-24 are currently pending and have been considered below.

Specification

2. The disclosure is objected to because of the following informalities:

- The specification contains the following typographical errors:
 - The word “can” in the sentence “the install controller *can* and/or the system under test ...” should be deleted in page 8, paragraph [0030].
 - The letter “p” in “perl” and the letters “a” and “s” in “applescript” should be capitalized in page 14, paragraph [0047].
- The specification does not explain what the acronyms MSI, MS, SKU, and MD5 stand for.

Appropriate correction is required.

3. The use of trademarks, such as MICROSOFT and APPLESCRIPT, has been noted in this application. Trademarks should be capitalized wherever they appear (capitalize each letter OR accompany each trademark with an appropriate designation symbol, e.g., ™ or ®) and be accompanied by the generic terminology (use trademarks as adjectives modifying a descriptive noun, e.g., “the MICROSOFT operating system”).

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner, which might adversely affect their validity as trademarks.

Claim Objections

4. **Claims 2, 9, 18, and 22** are objected to because of the following informalities:
 - **Claim 2** contains a typographical error: the phrase “resources that change” should presumably be read “resources that have changed” in the second limitation.
 - **Claims 9, 18, and 22** contain a typographical error: there should be a comma (,) before the word “and” to separate the last two elements in the series. Although the use of a comma before “and” in a series is not mandatory, applicant is advised to make the correction in order to keep the grammatical style consistent throughout the specification and claims.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.
6. **Claim 1** is rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The breadth of the claim relevant to enablement is too broad with

respect to the disclosure and would not enable one skilled in the art to make and use the entire scope of the claimed invention without undue experimentation.

Claim 1 is a single step claim, since the claimed method comprises of only one step. A single means/step claim, i.e., where a means/step recitation does not appear in combination with another recited element of means/step, is subject to an undue breadth rejection under 35 U.S.C. 112, first paragraph. *In re Hyatt*, 708 F.2d 712, 714-715, 218 USPQ 195, 197 (Fed. Cir. 1983) (A single means/step claim which covered every conceivable means/step for achieving the stated purpose was held nonenabling for the scope of the claim because the specification disclosed at most only those means/step known to the inventor.). When claims depend on a recited property, a fact situation comparable to *Hyatt* is possible, where the claim covers every conceivable structure (means) or step for achieving the stated property (result) while the specification discloses at most only those known to the inventor. See MPEP § 2164.08(a).

7. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

8. **Claim 22** is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 22 recites the acronym SKU. There is no explanation of what the acronym stands for in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “Stock Keeping Unit” for the purpose of further examination.

Claim Rejections - 35 USC § 101

9. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

10. **Claims 1-3, 10-12, and 24** are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

The claimed machine-implemented method of **Claim 1** is a “single step claim,” where the method consists of only one recited step. A claimed method with only one recited step is clearly not a “process” under § 101 because it is not a series of steps. Also, any known utility of the claimed invention cannot be realized from a method consisting of only one recited step. Therefore, the claimed invention does not fall within an enumerated statutory category of patentable subject matter (process, machine, manufacture, or composition of matter).

Furthermore, the result of **Claim 1** is directed to the act of “creating data,” which does not appear to have a readily apparent useful result so as to constitute a practical application of the idea. Apart from the utility requirement of 35 U.S.C. § 101, usefulness under the patent eligibility standard requires significant functionality to be present to satisfy the useful result aspect of the practical application requirement. See *Arrhythmia*, 958 F.2d at 1057, 22 USPQ2d at

1036. The mere fact that the claim may satisfy the utility requirement of 35 U.S.C. § 101 does not mean that a useful result is achieved under the practical application requirement. The claimed invention as a whole must produce a “useful, concrete, and tangible” result to have a practical application.

The result of **Claims 2, 3, 10, and 12** is directed to the act of “identifying,” which does not appear to be a tangible result so as to constitute a practical application of the idea. The act of “identifying” is merely a thought or an abstract idea and does not appear to produce a tangible result even if the step of identification does occur, since the result of that identification is not conveyed in the real world. The result is an identification, which is neither used in a disclosed practical application nor made available for use in a disclosed practical application. It also does not appear that the usefulness of the identification can be realized from the claimed steps to support a disclosed specific, substantial, and credible utility so as to produce a useful result.

In addition, the result of **Claim 11** is directed to the act of “designating,” which also does not appear to be a tangible result so as to constitute a practical application of the idea. The reasoning is stated above in the aforementioned paragraph and applied in the same manner.

Therefore, the claims do not meet the statutory requirement of 35 U.S.C. § 101, since the claims are not directed to a practical application of the § 101 judicial exception producing a result tied to the physical world.

It is noted that each of the means in **Claim 24** can be reasonably interpreted under 35 U.S.C. 112, sixth paragraph, as software alone. Therefore, **Claim 24** is directed to a system of

functional descriptive material *per se*, and hence non-statutory. The claim constitutes computer programs representing computer listings *per se*. Such descriptions or expressions of the programs are not physical “things.” They are neither computer components nor statutory processes, as they are not “acts” being performed. Such claimed computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer, which permit the computer program’s functionality to be realized. In contrast, a claimed computer-readable medium encoded with a computer program is a computer element, which defines structural and functional interrelationships between the computer program and the rest of the computer, that permits the computer program’s functionality to be realized, and is thus statutory. See *Lowry*, 32 F.3d at 1583-84, 32 USPQ2d at 1035.

Claim Rejections - 35 USC § 102

11. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

12. **Claims 1-6, 10, 12-15, and 24** are rejected under 35 U.S.C. 102(e) as being anticipated by Breggin et al. (US 6,560,776).

As per **Claim 1**, Breggin et al. disclose a machine-implemented method comprising:

A. Creating data that represents a new expectation for an installation result, for one or more resources associated with a software installer, the new expectation being a transition from an expectation of volatility to an expectation of stability for future software installs (*see Figure 1, Element 188; Column 1, Lines 62-65; Column 7, Lines 23-25, 29-46, and 51-67; and Column 8, Lines 1-23*).

As per **Claim 2**, Breggin et al. disclose a machine-implemented method **as in Claim 1 above**, and further disclose that the method comprising:

A. Generating a comparison of a current software installation with a previous software installation (*see Figure 3A, Element 228; Column 1, Lines 65-67; Column 2, Line 1; and Column 9, Lines 29-37*); and

B. Identifying, based on the comparison, resources that have changed in their installation result from the previous software installation to the current software installation, despite the new expectation of stability for the resources (*see Figure 5; Column 9, Lines 58-66; and Column 10, Lines 3-9 and 11-14*).

As per **Claim 3**, Breggin et al. disclose a machine-implemented method **as in Claim 2 above**, and further disclose that the method comprising identifying, based on the comparison, resources that have not changed in their installation result from the previous software installation to the current software installation, despite an expectation that the unchanged resources should

change from the previous software installation to the current software installation (*see Column 10, Lines 9-11 and 14-15*).

As per **Claim 4**, Breggin et al. disclose a machine-implemented method **as in Claim 2 above**, and further disclose that the method comprising presenting potential problems with the current software installation based on the identified resources to facilitate verification of an installer for a software product (*see Figure 3B, Element 236; Figure 5; Column 9, Lines 55-58; and Column 10, Lines 17-28*).

As per **Claim 5**, Breggin et al. disclose a machine-implemented method **as in Claim 4 above**, and further disclose that the method comprising tracking expectations for the resources in a primary installation baseline and a secondary installation baseline, and wherein presenting the potential problems comprises presenting a baseline-update interface by transmitting markup language data (*see Column 10, Lines 40-42 and 49-67; and Column 11, Lines 1-5*).

As per **Claim 6**, Breggin et al. disclose a machine-implemented method **as in Claim 4 above**, and further disclose that the method comprising excluding a set of resources from the generated comparison for the software product (*see Column 3, Lines 14-15; and Column 11, Lines 5-8*).

As per **Claim 10**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium, the software product comprising instructions operable to cause one or more data processing apparatus to perform operations comprising:

- A. Generating a comparison of a current software installation with a previous software installation (*see Figure 3A, Element 228; Column 1, Lines 65-67; Column 2, Line 1; and Column 9, Lines 29-37*); and
- B. Identifying based on the comparison, resources that have not changed in their installation result from the previous software installation to the current software installation, despite an expectation that the unchanged resources should change from the previous software installation to the current software installation (*see Figure 5; Column 9, Lines 58-66; and Column 10, Lines 3-9 and 11-14*).

As per **Claim 12**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium **as in Claim 10 above**, and further disclose that the operations comprise identifying, based on the comparison, resources that have changed in their installation result from the previous software installation to the current software installation, despite an expectation that the changed resources should not change from the previous software installation to the current software installation (*see Figure 5; Column 9, Lines 58-66; and Column 10, Lines 3-9 and 11-14*).

As per **Claim 13**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium **as in Claim 12 above**, and further disclose that the operations

comprise presenting potential problems with the current software installation based on the identified resources to facilitate verification of an installer for a software product (*see Figure 3B, Element 236; Figure 5; Column 9, Lines 55-58; and Column 10, Lines 17-28*).

As per **Claim 14**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium **as in Claim 13 above**, and further disclose that the operations comprise tracking the expectations of resource changes in a primary installation baseline and a secondary installation baseline, and wherein presenting the potential problems comprises presenting a baseline-update interface by transmitting markup language data (*see Column 10, Lines 40-42 and 49-67; and Column 11, Lines 1-5*).

As per **Claim 15**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium **as in Claim 13 above**, and further disclose that the operations comprise excluding a set of resources from the generated comparison for the software product (*see Column 3, Lines 14-15; and Column 11, Lines 5-8*).

As per **Claim 24**, the Applicant appears to be attempting to invoke 35 U.S.C. 112, sixth paragraph, since it contains “means-plus-function” limitations. However, the Examiner notes that the only “means” for performing these recited functions in the specification appears to be computer program modules—software *per se*. While the claim meets the first prong of the three-prong analysis used to determine invocation of 35 U.S.C. 112, sixth paragraph, the claim does not meet the other prongs of the three-prong analysis, since no other specific corresponding

structure or equivalents thereof are disclosed in the specification. Therefore, the claim limitations are not being treated under 35 U.S.C. 112, sixth paragraph.

Breggin et al. disclose a system comprising:

A. Means for generating a current install comparison of a computing system before a software installation, with the computing system after the software installation, the current install comparison identifying new resources that are added to the computing system during the software installation and identifying system resources that are modified during the software installation, and the current install comparison recording at least one attribute of the resources for the current software installation (*see Figure 3A, Element 228; Figure 5; Column 1, Lines 65-67; Column 2, Line 1; and Column 9, Lines 29-37*);

B. Means for generating a software trend comparison of the current install comparison with a previous install comparison, the software trend comparison including which of the resources have changed in the at least one attribute from the previous install to the current install, and the software trend comparison indicating which of the resources have not changed in the at least one attribute from the previous install to the current install (*see Figure 3A, Element 228; Figure 5; Column 1, Lines 65-67; Column 2, Line 1; and Column 9, Lines 29-37*);

C. Means for comparing the software trend comparison with a record of installation expectations that indicates which of the resources should be in flux, and which of the resources should be stable from the previous install to the current install, with respect to the at least one attribute (*see Figure 5; Column 9, Lines 58-66; and Column 10, Lines 3-15*); and

D. Means for presenting potential problems with the current software installation based on the comparison of the software trend comparison with the expectations record (*see Figure 3B, Element 236; Figure 5; Column 9, Lines 55-58; and Column 10, Lines 17-28*).

Claim Rejections - 35 USC § 103

13. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

14. **Claims 7-9 and 16-18** are rejected under 35 U.S.C. 103(a) as being unpatentable over Breggin et al. (US 6,560,776).

As per **Claim 7**, Breggin et al. disclose a machine-implemented method **as in Claim 4 above**, and further disclose that the expectations of resource changes, including the data, are stored in a relational database indexed by date (*see Column 7, Lines 47-49; and Column 8, Lines 24-29*). However, Breggin et al. does not explicitly disclose that the expectations of resource changes are indexed by platform, language, and product configuration.

Nevertheless, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate indexing the expectations of resource changes by platform, language, and product configuration in the system of Breggin et al., since Breggin et al. is already indexing the expectations of resource changes by date and other information, such as file

location and file version. One would have been motivated to index the expectations of resource changes by platform, language, and product configuration in order to store and access additional useful data in the database pertaining to the software installation.

As per **Claim 8**, Breggin et al. disclose a machine-implemented method **as in Claim 4 above**, and further disclose that the expectations for the resources relate to attributes comprising modification date stamp information and file size information (*see Column 8, Lines 24-29*). However, Breggin et al. does not explicitly disclose that the expectations for the resources relate to attributes comprising security permissions information and checksum information.

Nevertheless, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include security permissions information and checksum information as part of the attributes in the system of Breggin et al., since Breggin et al. already has attributes on modification date stamp information, file size information, and other information, such as file location and file version. One would have been motivated to include security permissions information and checksum information as attributes in order to provide additional useful data pertaining to the software installation to the user.

As per **Claim 9**, Breggin et al. disclose a machine-implemented method **as in Claim 4 above**, and further disclose that the resources comprise files and system registry entries (*see Column 7, Lines 23-25*) and the installation result comprises modifications of the resources (*see Column 10, Lines 61-64*). However, Breggin et al. does not explicitly disclose that the installation result comprises deletions and additions of the resources.

Nevertheless, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include deletions and additions of the resources in the system of Breggin et al., since Breggin et al. already has modifications of the resources in the installation result. One would have been motivated to include deletions and additions in the installation result in order to provide additional installation information that can be used for system monitoring.

As per **Claim 16**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium **as in Claim 13 above**, and further disclose that the expectations of resource changes, including the data, are stored in a relational database indexed by date (*see Column 7, Lines 47-49; and Column 8, Lines 24-29*). However, Breggin et al. does not explicitly disclose that the expectations of resource changes are indexed by platform, language, and product configuration.

Nevertheless, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate indexing the expectations of resource changes by platform, language, and product configuration in the system of Breggin et al., since Breggin et al. is already indexing the expectations of resource changes by date and other information, such as file location and file version. One would have been motivated to index the expectations of resource changes by platform, language, and product configuration in order to store and access additional useful data in the database pertaining to the software installation.

As per **Claim 17**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium **as in Claim 13 above**, and further disclose that the expectations of

resource changes relate to attributes comprising modification date stamp information and file size information (*see Column 8, Lines 24-29*). However, Breggin et al. does not explicitly disclose that the expectations of resource changes relate to attributes comprising security permissions information and checksum information.

Nevertheless, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include security permissions information and checksum information as part of the attributes in the system of Breggin et al., since Breggin et al. already has attributes on modification date stamp information, file size information, and other information, such as file location and file version. One would have been motivated to include security permissions information and checksum information as attributes in order to provide additional useful data pertaining to the software installation to the user.

As per **Claim 18**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium **as in Claim 13 above**, and further disclose that the resources comprise files and system registry entries (*see Column 7, Lines 23-25*) and the installation result comprises modifications of the resources (*see Column 10, Lines 61-64*). However, Breggin et al. does not explicitly disclose that the installation result comprises deletions and additions of the resources.

Nevertheless, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include deletions and additions of the resources in the system of Breggin et al., since Breggin et al. already has modifications of the resources in the installation result.

One would have been motivated to include deletions and additions in the installation result in order to provide additional installation information that can be used for system monitoring.

15. **Claims 11, 19, 20, 22, and 23** are rejected under 35 U.S.C. 103(a) as being unpatentable over Breggin et al. (US 6,560,776) in view of Kruger et al. (US 6,738,970).

As per **Claim 11**, Breggin et al. disclose a software product tangibly embodied in a machine-readable medium **as in Claim 10 above**, and further disclose that the operations comprise designating a new expectation of stability for the specified resources according to the received input (*see Figure 1, Element 188; Column 1, Lines 62-65; Column 7, Lines 23-25, 29-46, and 51-67; and Column 8, Lines 1-23*). However, Breggin et al. does not explicitly disclose that the operations comprise receiving input specifying which of the identified resources should be static in their installation result for future software installations.

In the same field of endeavor, Kruger et al. disclose a method and apparatus identifies changes made to a computer system caused by the installation of software, where trees are built to record the state of the computer system before and after the user installs the computer software onto his computer system. Tree nodes that are marked as “same” are removed so the objects corresponding to these nodes will not need to be altered during subsequent installations (*see Column 12, Lines 37-40*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to allow receiving input specifying which of the identified resources should be static in their installation result for future software installations in the system of Breggin et al.,

since Breggin et al. already allows tracking resources that have changed (*see Column 10, Lines 61-64*). One would have been motivated to allow receiving input specifying which of the identified resources should be static in their installation result for future software installations in order to prevent errors from occurring during subsequent software installations.

As per **Claim 19**, Breggin et al. disclose a system comprising:

- A. A build controller (*see Column 3, Lines 66-67; and Column 4, Lines 1-6*); and
- B. An install controller comprising a database including a baseline recording

expectations of stability or volatility for one or more resources associated with a software installer (*see Figure 1, Element 200; Column 7, Lines 47-49; and Column 10, Lines 50-52*);

Wherein the build controller automatically triggers the install controller to initiate installer tests as part of a software build process (*see Column 4, Lines 16-21*), and collects test results to be presented in a report comprising a baseline-update interface (*see Figure 3B, Element 236; Figure 5; Column 9, Lines 55-58; and Column 10, Lines 17-28*).

However, Breggin et al. does not explicitly disclose that the system comprising one or more install slave machines, and the install controller automatically dispatches installation to the one or more install slave machines.

In the same field of endeavor, Kruger et al. disclose a method and apparatus identifies changes made to a computer system caused by the installation of software, where trees are built to record the state of the computer system before and after the user installs the computer software onto his computer system. The master computer is any computer on which the computer software can be properly installed, and for which such installation will be used as a model for

installation of the software on other computer systems (*see Column 4, Lines 1-5*). Also, the system sends the instructions, files, and program to other computer systems using conventional management software (*see Column 4, Lines 19-27*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate one or more install slave machines and automatically dispatches installation to the slave machines in the system of Breggin et al., since Breggin et al. is already utilizing a master database (*see Figure 4*). One would have been motivated to incorporate one or more install slave machines and automatically dispatches installation to the slave machines in order to provide redundant data backup or testing platforms for diagnosing and monitoring software installation/performance.

As per **Claim 20**, Breggin et al., as modified by Kruger et al., disclose a system as in **Claim 19 above**, but Breggin et al. does not explicitly disclose that the one or more slave machines comprise multiple computers. However, Kruger et al. does disclose that one or more slave machines comprise multiple computers (*see Column 4, Lines 1-5*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate multiple computers as slave machines in the system of Breggin et al., since software installations are performed on computer systems in Breggin et al.. One would have been motivated to use multiple computers as slave machines in order to properly install and test software programs.

As per **Claim 22**, Breggin et al., as modified by Kruger et al., disclose a system as in **Claim 19 above**, and Breggin et al. further disclose that the baseline-update interface comprises a web-based user interface allowing baseline updates across product Stock Keeping Unit (SKU), language, operating system, and custom/non-custom installs, in combination or all at once (*see Column 10, Lines 40-42*).

As per **Claim 23**, Breggin et al., as modified by Kruger et al., disclose a system as in **Claim 19 above**, and Breggin et al. further disclose that the expectations of resource changes relate to attributes comprising modification date stamp information and file size information (*see Column 8, Lines 24-29*). However, Breggin et al. and Kruger et al. do not explicitly disclose that the expectations of resource changes relate to attributes comprising security permissions information and checksum information.

Nevertheless, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include security permissions information and checksum information as part of the attributes in the system of Breggin et al., since Breggin et al. already has attributes on modification date stamp information, file size information, and other information, such as file location and file version. One would have been motivated to include security permissions information and checksum information as attributes in order to provide additional useful data pertaining to the software installation to the user.

16. **Claim 21** is rejected under 35 U.S.C. 103(a) as being unpatentable over Breggin et al. (US 6,560,776) in view of Kruger et al. (US 6,738,970) as applied to Claim 19 above, and further in view of Suorsa et al. (US 2002/0156831).

As per **Claim 21**, Breggin et al., as modified by Kruger et al., disclose a system as in **Claim 19 above**, but neither reference discloses that the install controller communicates with the one or more install slave machines using Simple Object Access Protocol.

However, in the same area the problem sought to be solved, Suorsa et al. disclose a similar system that exchanges messages between the gateway and the agents in the form of remote procedure calls that conform to the XML-RPC protocol, or the Simple Object Access Protocol (SOAP) (*see Paragraph [0052]*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to communicate with the one or more install slave machines using Simple Object Access Protocol (SOAP) in the system of Breggin et al., since Breggin et al. disclose that the installed database or file can be incorporated into one or more web pages in Web-based applications (*see Column 10, Lines 40-42*). One would have been motivated to communicate with the one or more install slave machines using Simple Object Access Protocol (SOAP) in order to provide a way to communicate between applications running on different operating systems with different technologies and programming languages.

Conclusion

17. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

A. Goossen et al. (US 6,071,316) disclose a method and apparatus for automating validation and verification of computer software that confirms during a test execution of the software that all lines of code are executed and all branches in the software are taken or not taken at least once.

B. Sliger et al. (US 6,216,175) disclose updating of computer software by use of patch files, the generation of such patch files, and the normalization of files to which such patches are applied.

C. Valys et al. (US 6,535,915) disclose a method for reduction of data noise in installation packages for a computer system.

D. Luu (US 6,944,858) discloses a technique for the remote installation of application software from a source computer system to one or more target computer systems (workstations) coupled to a Local Area Network (LAN).

E. Singleton (US 6,978,454) discloses a computer program tool for checking the correct or otherwise installation of a different computer program operates by comparing predetermined installation data with detected installation results, including at least one of read register entries and details of files stored subsequent to the installation.

F. McMillan et al. (US 7,028,019) disclose a system and method for managing software conflicts and computer-readable storage medium having a program for executing the method wherein a database of interrelated tables is utilized.

G. Simpson (US 2003/0051235) discloses a method for verifying and analyzing the installation of software onto a programmable device.

H. Gartside et al. (US 2003/0192033) disclose validation of the installation of a computer program on a computer.

I. Sheehy (US 2004/0243995) discloses a system and method for managing installs to a set of one or more field machines in a distributed network environment.

J. Loh et al. (US 2006/0075399) disclose a system and method for predicting a quantity of a resource required for the deployment of a software application on a computing system.

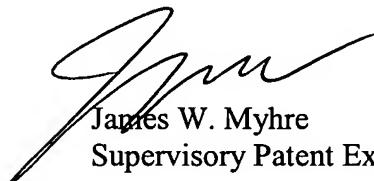
Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, James W. Myhre, can be reached on 571-270-1065. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR

system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

QC / QC
June 21, 2006



James W. Myhre
Supervisory Patent Examiner